On the word problem for SP-categories, and the properties of two-way communication

Luigi Santocanale, Robin Cockett

On the word problem for $\Sigma\Pi$-categories, and the properties of two-way communication[⋆] [⋆⋆]

Robin Cockett[1] and Luigi Santocanale[2]

[1] Department of Computer Science,
University of Calgary
robin@ucalgary.ca
[2] Laboratoire d'Informatique Fondamentale de Marseille,
Université Aix-Marseille I,
luigi.santocanale@lif.univ-mrs.fr

Abstract. The word problem for categories with free products and coproducts (sums), $\Sigma\Pi$-categories, is directly related to the problem of determining the equivalence of certain processes. Indeed, the maps in these categories may be directly interpreted as processes which communicate by two-way channels.
The maps of an $\Sigma\Pi$-category may also be viewed as a proof theory for a simple logic with a game theoretic intepretation. The cut-elimination procedure for this logic determines equality only up to certain permuting conversions. As the equality classes under these permuting conversions are finite, it is easy to see that equality between cut-free terms (even in the presence of the additive units) is decidable. Unfortunately, this does not yield a tractable decision algorithm as these equivalence classes can contain exponentially many terms.
However, the rather special properties of these free categories – and, thus, of two-way communication – allow one to devise a tractable algorithm for equality. We show that, restricted to cut-free terms $s, t : X \to A$, the decision procedure runs in time polynomial on $|X| \cdot |A|$, the product of the sizes of the domain and codomain type.
Keywords. $\Sigma\Pi$-categories, bicatersian categories, word problem, two-way communication, game semantics.

Introduction

We present a decision procedure for equality of parallel arrows in $\Sigma\Pi$-categories. These categories have (chosen) finite sums (coproducts) and finite products, including, significantly, the units for these categorical operations. Thus, the categories we consider do have an initial object, the unit for the sum, and a terminal object, the unit for the product.

Recall that word problems for algebraic theories amount to studying the free models of these theories. Here the situation is analogous: the theory of $\Sigma\Pi$-categories – being an essentially algebraic theory – has free models; the decision

[⋆] Extended abstract.
[⋆⋆] Research partially supported by the ANR project SOAPDC no. JC05-57373.

procedure we describe relies crucially on a number of algebraic facts peculiar to free $\Sigma\Pi$-categories.

While the categorical structure we are investigating is one of the simplest, the status of the word problem for these categories has languished in an unsatisfactory state. It is decidable as standard tools from categorical logic [1,2] allow free $\Sigma\Pi$-categories to be viewed as deductive systems for logics. In [3] these deductive systems were shown to correspond precisely to the usual categorical coherence requirements for products and sums and, furthermore, to satisfy the cut-elimination property. The focus of the decision procedure then devolves upon the cut-free terms whose equivalence is completely determined by a finite number of "permuting conversions".

The cut-free terms, which represent arrows between two given types, are finite in number and this implies, immediately, that equality is decidable. However, the implied complexity of this way of deciding equality is exponential because there can be an exponential number of equivalent terms. The question, which still remained open, was whether the matter could be decided in polynomial time. This was of particular interest as these expressions are, in the process world, the analogue of Boolean expressions. The main contribution of this paper is to confirm that there is a polynomial algorithm which settles this question.

There have been, directly or indirectly, a number of contributions towards our goal in this paper. Most of them involve a representation theorem, that is the provision of a full and faithful functor from some variant of the free $\Sigma\Pi$-category into a concrete combinatoric category. For example [4] considers $\Sigma\Pi$-categories, in which the initial and final object coincide and represents these using a subcategory of the category of coherent spaces, while [5] and [6] both gives a representation of $\Sigma\Pi$-categories *without units* into, respectively, a combinatoric category of proof-nets and the category of sets and relations. These related results, however, work only for the fragment without units – or, more precisely, for the fragment with a common initial and final object. As far we know, there is no representation theorem for the full fragment with distinct units.

Units add to the decision problem – and to the representation theory – a non-trivial challenge which is easy to under estimate. In particular, in [3], one of the current authors was guilty of rather innocently proposing an altogether too simple decision procedure which, while working perfectly in the absence of units, fails manifestly in the presence of units. The effect of the presence of units on the setting is quite dramatic. In particular, when there are no units (or there is a zero) *all* coproduct injections are monic. However, rather contrarily, in the presence of distinct units this simply is not longer the case. Furthermore, this can be demonstrated quite simply, consider the following diagram:

$$\begin{tikzcd} 0 \times 0 \arrow[r, shift left, "\pi_1"] \arrow[r, shift right, "\pi_0"'] & 0 \arrow[r, "\sigma_0"] & 0+1 \end{tikzcd}$$

As $0 + 1 \simeq 1$ is a terminal object, there is at most one arrow to it: this makes the above diagram a coequalizer. Yet, the arrows π_0, π_1 are distinct in a free $\Sigma\Pi$-category, as for example they receive distinct interpretations in the dual category of the category of sets and functions.

As logicians and category theorists, we were deeply frustrated by this failure to master the units. The solution we now present for this decision problem, however, was devised only after a much deeper algebraic understanding of the structure of free $\Sigma\Pi$-categories had been obtained. The technical observations which underly this development, we believe, should be of interest to logician and category theorists alike. Yet, our principal motivation for studying the theory of $\Sigma\Pi$-categories and free $\Sigma\Pi$-categories arose from the role they have as models of computation. We discuss in details this point next.

The proof theory of (free) sums and products in a category is remarkable from a number of points of view. Not only does it provide an elegant proof theory with a rich underlying algebraic calculus, but also it supports a variety of quite surprising interpretations.

The most immediate interpretation, but by no means one which is transparently obvious see [7,8,9], is as a game theory in which the types represent finite games. The products have the role of opponent while the sums have the role of player and there is no requirement that plays alternate. The maps are then interpreted as being mediators between games which use the information of one game to determine the play on the other. Their composition is given by hiding the transfer of information which happens through moves on a middle game.

Proof theoretically this composition can be viewed as a cut-elimination process which, in turn, algebraically translates into an elegant reduction system which is confluent modulo equations (the details are to be found in [3]). As shall become clear this paper is largely concerned with the consequences of the equations which remain after the cut has been eliminated. However, before discussing this we must describe a second important and appealing interpretation.

Arrows in the free category with sums and products can also be interpreted as processes which communicate along channels: the types are the (finite) protocols which govern the interactions along these channels. These protocols tell a process which wishes to communicate along a channel whether it is the turn of the process to send a message (and precisely which messages can then be sent) or whether it is the turn of the process to listen (and precisely which messages can be received). This is more than an idle idea: the theoretical details of this interpretation have been fleshed out in some detail (and, in fact, more generally to allow multiple channels) in [10,11].

This last interpretation is quite compelling as the algebraic results described in this paper suggest a number of not very obvious and even somewhat surprising properties of communication along a channel. For example, a process which is required to send a value could send various different values and yet, semantically, remain *exactly the same process*. This is the notion of *indefiniteness* which is central in the business of unraveling the meaning of communication. There are various situations in which this apparently unintuitive situation can arise. For example, it could be that the recipient of the communication has simply stopped listening. It is of course very annoying when this happens but, undeniable, this is an occurrence well within the scope of the human experience of communication. However, it can also be, more dramatically, that the sender has stopped communicating to the receiver – and this produces what we shall call a *disconnect*.

Proof theoretically and algebraically this all has to do with the behavior of the (additive) unit, that is, the final object and the initial object. The purpose of this paper is to focus on these units and their ramifications in the whole business of communication. It is certainly true that without the units the situation is very much simpler. However, if one is tempted therefore simply to omit them, it is worth realizing that without any units there is simply no satisfactory notion of *a finite communication*!

Of course, without the units the theory is not only simpler but a good deal less mathematically interesting. It is this mathematics which we now turn to.

The paper is structured as follows. In Section 1, we recall the elementary definition of $\Sigma\Pi$-category, and the results of [3]. In Section 2, we start our analysis of the main property of $\Sigma\Pi$-categories, softness, in order to give it a more concrete meaning, accessible to the general logician, in terms of a sort of undirected rewrite system. In Section 3, we shall present our first main result, stating that coproduct injections are weakly disjoint in free $\Sigma\Pi$-categories, and list some consequences. This leads to a discussion of arrows which factor through a unit – indefinite arrows – which play a key role in the decision procedure. In Section 4 we present our second main observation: if two arrows in $\hom(X\times Y, A)$ and $\hom(Y, A+B)$ are definite but are made equal when, respectively, projecting and coprojecting into $\hom(X \times Y, A + B)$, this fact is witnessed by a unique "bouncer" in $\hom(Y, A)$. In Section 5, we collect our observations and sketch the decision procedure.

1 The construction of free $\Sigma\Pi$-categories

1.1 $\Sigma\Pi$-categories

We invite the reader to consult [12] for the basic categorical notions used in this paper. Here, an *$\Sigma\Pi$-category* shall mean a category with finite products and finite coproducts.

Recall that a category has *binary products* if, given two objects A, B, there exists a third object $A \times B$, and natural transformations

$$\hom(X, A) \times \hom(X, B) \xrightarrow{\langle\, ,\, \rangle} \hom(X, A \times B)\,,$$
$$\hom(X_i, A) \xrightarrow{\pi_i} \hom(X_0 \times X_1, A)\,,\ i = 0, 1\,,$$

that induce inverse bijections:

$$\pi_i(\langle f_0, f_1\rangle) = f_i\,,\ i = 0, 1\,, \qquad\qquad \langle \pi_0(f), \pi_1(f)\rangle = f\,.$$

A *terminal object* or *empty product* in a category is an object 1 such that, for each object X, $\hom(X, 1)$ is a singleton. It is part of standard theory that a terminal object is unique up to isomorphism and that it is the unit for then product, as $X \times 1$ is canonically isomorphic to X.

We obtain the definition of binary sums (or coproducts) and of initial object, by exchanging the roles of left and right objects in the definition of products: a category has *binary sums* if, given two objects X, Y, there exists a third object $X + Y$ and natural transformations

$$\mathrm{hom}(X, A) \times \mathrm{hom}(Y, A) \xrightarrow{\{\ ,\ \}} \mathrm{hom}(X + Y, A)\,,$$
$$\mathrm{hom}(X, A_j) \xrightarrow{\sigma_j} \mathrm{hom}(X, A_0 + A_1)\,,\ j = 0, 1\,,$$

that induce inverse bijections:

$$\sigma_j(\{f_0, f_1\}) = f_j\,,\ j = 0, 1\,, \qquad\qquad \{\sigma_0(f), \sigma_1(f)\} = f\,.$$

An *initial object* 0 is such that, for each object A, $\mathrm{hom}(0, A)$ is a singleton.

$$\begin{tikzcd} \mathbb{A} \arrow[r, "\eta"] \arrow[dr, "F"'] & \Sigma\Pi(\mathbb{A}) \arrow[d, dotted, "\exists!\,\tilde{F}"] \\ & \mathbb{B} \end{tikzcd}$$

A functor between two $\Sigma\Pi$-categories $\mathbb{A}, \mathbb{B}$ is a *$\Sigma\Pi$-functor* if it sends (chosen) products to products, and (chosen) coproducts to coproducts. The *free $\Sigma\Pi$-category over a category* $\mathbb{A}$, denoted $\Sigma\Pi(\mathbb{A})$, has the following property: there is a functor $\eta : \mathbb{A} \to \Sigma\Pi(\mathbb{A})$ such that, if $F : \mathbb{A} \to \mathbb{B}$ is a functor that "interprets" $\mathbb{A}$ into a $\Sigma\Pi$-category $\mathbb{B}$, then there exists a unique $\Sigma\Pi$-functor $\tilde{F} : \Sigma\Pi(\mathbb{A}) \to \mathbb{B}$ such that $\tilde{F} \circ \eta = F$. This is the usual universal property illustrated by the diagram on the right.

The free $\Sigma\Pi$-category on $\mathbb{A}$, can be "constructed" as follows. Its objects are the types inductively defined by the grammar

$$T = \eta(x) \mid 1 \mid T \times T \mid 0 \mid T + T\,, \tag{1}$$

where x is an object of $\mathbb{A}$. Then proof-terms are generated according to the deduction system of figure 1. Finally, proof-terms $t : X \to A$ are quotiented by means of the least equivalence relation that forces the equivalence classes to satisfy the axioms of a $\Sigma\Pi$-category. Of course, while this is a perfectly good specification, we are looking for an effective presentation for $\Sigma\Pi(\mathbb{A})$. A first step in this direction comes from the fact the identity-rule as well as the cut-rule can be eliminated from the system. More precisely we have the following theorem:

Proposition 1 (See [3] Proposition 2.9). *The cut-elimination procedure gives rise to a rewrite system that is confluent modulo the set of equations of figure 2.*

From this we obtain an effective description of the category $\Sigma\Pi(\mathbb{A})$: the objects are the types generated by the grammar (1), while the arrows are equivalence classes of (identity|cut)-free proof-terms under the least equivalence generated by the equations of figure 2. Composition is given by the cut-elimination procedure, which by the above theorem is well defined on equivalence classes.

Thus, our main goal in the rest of the paper is the following: given two proof-terms $s, t : X \to A$, are they equivalent according to the least equivalence relation generated by the equations of figure 2? The problem is easily seen to be decidable:

$$\frac{}{X \xrightarrow{id_X} X}\ \text{identity-rule} \qquad \frac{X \xrightarrow{f} C \quad C \xrightarrow{g} A}{X \xrightarrow{f;g} A}\ \text{cut-rule}$$

$$\frac{x \xrightarrow{f} y}{\eta(x) \xrightarrow{\eta(f)} \eta(y)}\ \text{Generators rule}$$

$$\frac{-}{X \xrightarrow{!} 1}\ R1$$

$$\frac{X_i \xrightarrow{f} A}{X_0 \times X_1 \xrightarrow{\pi_i(f)} A}\ L_i\times \qquad \frac{X \xrightarrow{f} A \quad X \xrightarrow{g} B}{X \xrightarrow{\langle f,g\rangle} A \times B}\ R\times$$

$$\frac{-}{0 \xrightarrow{?} A}\ L0$$

$$\frac{X \xrightarrow{f} A \quad Y \xrightarrow{g} A}{X + Y \xrightarrow{\{f,g\}} A}\ L+ \qquad \frac{X \xrightarrow{f} A_j}{X \xrightarrow{\sigma_j(f)} A_0 + A_1}\ R_j+$$

Fig. 1. The deductive system for $\Sigma\Pi(\mathbb{A})$

the contribution of this paper is to show that, furthermore, there is a feasible algorithm.

The main theoretical tool we shall use in developing this algorithm is the idea of *softness* which we now introduce. In every $\Sigma\Pi$-category there exist canonical maps

$$\begin{aligned}\coprod_j \hom(X, A_j) &\longrightarrow \hom(X, \coprod_j A_j)\,,\\ \coprod_i \hom(X_i, A) &\longrightarrow \hom(\prod_i X_i, A)\,.\end{aligned} \tag{2}$$

We shall be interested in these maps when, in a free $\Sigma\Pi$-category $\Sigma\Pi(\mathbb{A})$, $X = \eta(x)$ and $A = \eta(a)$ are generators.

$$\pi_i(\langle f, g\rangle) = \langle \pi_i(f), \pi_i(g)\rangle \qquad \sigma_j(\{f, g\}) = \{\sigma_j(f), \sigma_j(g)\}$$

$$\pi_i(\sigma_j(f)) = \sigma_j(\pi_i(f))$$

$$\{\langle f_{11}, f_{12}\rangle, \langle f_{21}, f_{22}\rangle\} = \langle\{f_{11}, f_{21}\}, \{f_{12}, f_{22}\}\rangle$$

$$\pi_i(!) = ! \qquad\qquad \sigma_j(?) = ?$$

$$\{!, !\} = ! \qquad\qquad \langle ?, ?\rangle = ?$$

$$!_0 = ?_1$$

Fig. 2. The equations on (identity|cut)-free proof-terms

In every $\Sigma\Pi$-category there also exist canonical commuting diagrams of the form

$$\begin{tikzcd}
\coprod_{i,j} \hom(X_i, A_j) \arrow[r] \arrow[d] & \coprod_j \hom(\prod_i X_i, A_j) \arrow[d] \\
\coprod_i \hom(X_i, \coprod_j A_j) \arrow[r] & \hom(\prod_i X_i, \coprod_j A_j)
\end{tikzcd} \tag{3}$$

The following key theorem holds:

Theorem 2 (See [3] Theorem 4.8). *The following properties hold of $\Sigma\Pi(\mathbb{A})$:*

1. *The functor $\eta : \mathbb{A} \to \Sigma\Pi(\mathbb{A})$ is full and faithful.*
2. *Generators are* atomic*, that is, the canonical maps of (2) – with $X = \eta(x)$ and $A = \eta(a)$ – are isomorphisms.*
3. *$\Sigma\Pi(\mathbb{A})$ is* soft*, meaning that the canonical diagrams of (3) are pushouts.*

Moreover, if $\mathbb{B}$ is a $\Sigma\Pi$-category with a functor $F : \mathbb{A} \to \mathbb{B}$, so that the pair $(F, \mathbb{B})$ satisfies 1,2,3, then the extension $\hat{F} : \Sigma\Pi(\mathbb{A}) \to \mathbb{B}$ is an equivalence of categories.

Thus, the structure of the category $\Sigma\Pi(\mathbb{A})$ is precisely determined by the conditions 1,2,3. We shall spend the next section giving an explicit account of the property of softness. The theorem is a special instance of the more general observations due to Joyal on free bicomplete categories [13,14].

2 An account of softness

A decision procedure necessarily focuses on the homset $\hom(X_0 \times X_1, A_0 + A_1)$ which, by Theorem 2, is a certain the pushout. Equivalently, this homset is the colimit of what we shall refer to as the "diagram of cardinals":

$$\begin{tikzcd}
\hom(X_0 \times X_1, A_0) & \hom(X_0, A_0) \arrow[l, "\pi_0"'] \arrow[r, "\sigma_0"] & \hom(X_0, A_0 + A_1) \\
\hom(X_1, A_0) \arrow[u, "\pi_1"] \arrow[d, "\sigma_0"'] & & \hom(X_0, A_1) \arrow[u, "\sigma_1"'] \arrow[d, "\pi_0"] \\
\hom(X_1, A_0 + A_1) & \hom(X_1, A_1) \arrow[l, "\sigma_1"'] \arrow[r, "\pi_1"] & \hom(X_0 \times X_1, A_1)
\end{tikzcd}$$

The explicit way of constructing such a colimit – see [12, §V.2.2] – is to first consider the sum S of the corners:

$$\hom(X_0, A_0 + A_1) + \hom(X_1, A_0 + A_1) + \hom(X_0 \times X_1, A_0) + \hom(X_0 \times X_1, A_1)$$

and then quotient S by the equivalence relation generated by elementary pairs, i.e. pairs (f, g) such that, for some h, $f = \pi_i(h)$ and $\sigma_j(h) = g$, as sketched below:

$$\begin{tikzcd}
& h \in \hom(X_i, A_j) \arrow[dl, "\pi_i"'] \arrow[dr, "\sigma_j"] & \\
f \in \hom(X_0 \times X_1, A_j) & & g \in \hom(X_i, A_0 + A_1)
\end{tikzcd}$$

Thus, for $f, f' \in S$ we have that $[f] = [f'] \in \hom(X_0 \times X_1, A_0 + A_1)$ if and only if there is a path in the diagram of cardinals from f to f', that is a sequence $f_0 f_1 f_2 \ldots f_n$, where $f = f_0$ and $f_n = f'$, such that, for $i = 0, \ldots, n-1$, (f_i, f_{i+1}) or (f_{i+1}, f_i) is an elementary pair.

3 The geometry of softness: weak disjointeness

Let us recall that a *point* in a $\Sigma\Pi$-category is an arrow of the form $p : 1 \to A$. When an object has a point we shall say it is *pointed*. Similarly, a *copoint* is an arrow of the form $c : X \to 0$ and an object with a copoint is *copointed*.

An object of $\Sigma\Pi(\emptyset)$ can be viewed as a two-player game on a finite tree, with no draw final position. Points then correspond then to winning strategies for the player, while copoints correspond to winning strategies for the opponent. Thus, by determinacy, every object of $\Sigma\Pi(\emptyset)$ either has a point or a copoint but not both.

The first important result for analyzing softness concerns copoints and coproduct injections:

Theorem 3. *Coproducts are, in $\Sigma\Pi(\mathbb{A})$, weakly disjoint: if $f;\sigma_0 = g;\sigma_1 : X \to A+B$, then there exists a copoint $c : X \to 0$ such that $f = c;?$ and $g = c;?$.*

$$\begin{tikzcd} X \arrow[rr, "g"] \arrow[dd, "f"'] \arrow[dr, dotted, "\exists c"] & & A_1 \arrow[dd, "\sigma_1"] \\ & 0 \arrow[ur, "?_{A_1}"] \arrow[dl, "?_{A_0}"] & \\ A_0 \arrow[rr, "\sigma_0"] & & A_0 + A_1 \end{tikzcd}$$

The property is illustrated in the diagram. The Theorem has an interesting interpretation from the perspective of processes: a process can send incoherent messages – white noise – on a channel without changing the meaning of the communication when and only when the recipient has stopped listening. The consequences of misjudging when the recipient stops listening, of course, is well-understood by school children and adults alike!

Proof. We sketch here the proof of the Theorem 3, emphasizing its geometrical flavor, as the diagram of cardinals is a sort of a one dimensional sphere.

We say that a triple $(X \mid A_0, A_1)$ is good if for every $f : X \to A_0$ and $g : X \to A_1$ the statement of the Theorem holds. Similarly, we say that a triple $(X_0, X_1 \mid A)$ is good if, for every $f : X_0 \to A$ and $g : X_1 \to A$, the dual statement of the Theorem holds. We prove that every triple is good, by induction on the structural complexity of a triple.

The non trivial induction step arises when considering a triple of the form $(X_0 \times X_1 \mid A_0, A_1)$ – or the dual case. Here, saying that the equality $f;\sigma_0 = g;\sigma_1$ holds means that there exists a path ϕ of the form $f_0 f_1 \ldots f_n$ in the diagram of cardinals from $f = f_0 \in \hom(X_0 \times X_1, A_0)$ to $g = f_n \in \hom(X_0 \times X_1, A_1)$, i.e. from northwest to southeast. Moreover, we may assume ϕ to be simple.

Such path necessarily crosses one of southwest or northeast corners, let us say the latter. This means that, for some $i = 1, \ldots, n-1$, $f_i \in \hom(X_0, A_0 + A_1)$, and f_{i-1}, f_{i+1} are in opposite corners. W.l.o.g. we can assume $f_{i-1} \in \hom(X_0 \times X_1, A_0)$ and $f_{i+1} \in \hom(X_0 \times X_1, A_1)$. Taking into account the definition of an elementary pair, we see that for some $h \in \hom(X_0, A_0)$ and $h' \in \hom(X_0, A_1)$ we have $h;\sigma_0 = f_i = h'\sigma_1$. Thus, by the inductive hypothesis on $(X_0 \mid A_0, A_1)$, we have $h = c;?_{A_0}$ and $h' = c;?_{A_1}$; in particular the projection $\pi_0 : X_0 \times X_1 \to X_0$ is epic, because of the existence of a copoint $c : X_0 \to 0$. Recalling that the path ϕ is simple, we deduce that i is the only time ϕ visits northeast, i.e. such that $f_i \in \hom(X_0, A_0 + A_1)$.

A similar analysis shows that if ϕ crosses a corner, then it visits that corner just once. Thus, we deduce that ϕ does not cross the northwest corner, as ϕ visits the northwest corner at time 0 and a corner may be crossed only at time $i \in \{ 1, \ldots, n-1 \}$. Similarly, ϕ does not cross the southeast corner. Also, ϕ cannot visit the southwest corner, as this would imply that at least one of northwest or southeast corners has been crossed.

Putting these considerations together, we deduce that ϕ visits the northwest, northeast, and southeast corners exactly once. That is, ϕ has length 2 and $i = 1$. Recalling the definition of elementary pair, we have

$$f = f_0 = \pi_0; h\,, \qquad h;\sigma_0 = f_1\,, \qquad f_1 = h';\sigma_1\,, \qquad \pi_0; h' = f_2 = g\,.$$

Considering that $h = c; ?_{A_0}$ and $h' = c; ?_{A_1}$, we deduce that $f = \pi_0; c; ?_{A_0}$ and $g = \pi_0; c; ?_{A_1}$. □

There are a number of consequences of this Theorem relevant to the decision procedure. To this end we need to introduce some terminology and some observations. We say that an arrow f is *pointed* if it factors through a point, i.e. if $f =!; p$ for some point p. Similarly, an arrow is *copointed* if it factors through a copoint. Note that an object A is pointed iff $? : 0 \to A$ is pointed and, similarly, X is copointed iff $! : X \to 1$ is copointed. A map which is neither pointed now copointed is said to be *definite*, otherwise it is said to be *indefinite*.

The following two facts are consequences of the theorem which can be obtained by a careful structural analysis:

Corollary 4.

1. *It is possible to decide (and find witnesses) in linear time in the size of a term whether it is pointed or copointed.*
2. *A coproduct injection $\sigma_0 : A \longrightarrow A + B$ is monic iff either B is not pointed or A is pointed. In particular $? : 0 \longrightarrow B$ is monic iff B is not pointed.*

An arrow is a *disconnect* if it is both pointed and copointed: it is easy to see that there is at most one disconnect between any two objects. Furthermore, if an arrow $f : A \to B$ is copointed, that is $f = c; ?$, and its codomain, B, is pointed then f is this unique disconnect. On the other hand, if the codomain B is not pointed then $? : 0 \to B$ is monic and, thus, such an f corresponds precisely to the copoint c. These observations allow the equality of indefinite maps, i.e. pointed and copointed, to be decided in linear time.

A further important fact which also follows from 4, in a similar vein to the above, concerns whether a map in $\Sigma\Pi(\mathbb{A})$ factors through a projection or a coprojection. This can also be decided in linear time on the size of the term. This is by a structural analysis which we now sketch.

Suppose that we wish to determine whether $f = \sigma_0(f') : A \to B + C$. If syntactically f is $\sigma_1(f')$ then, as a consequence of Theorem 3, the only way it can factorize is if the map is copointed. However, whether f is copointed can be determined in linear time on the term by Corollary 4. The two remaining possibilities are that f is syntactically $\{f_1, f_2\}$ or $\pi_i(f')$. In the former case, inductively, both f_1 and f_2 have to factorize through σ_0. In the latter case, when the map is not copointed, f' itself must factorize through σ_0.

There is, at this point, a slight algorithmic subtelty: to determine whether f can be factorized through a projection it seems that we may have to repeatedly recalculate whether the term is pointed or copointed and this recalculation would, it seems, push us beyond linear time. However, it is not hard to see that this the recalculation can be avoided simply by processing the term initially to include this information into the structure of the term (minimally two extra bits are needed at each node to indicate pointedness and copointedness of the map): subsequently this information would be available at constant cost. The cost of adding this information into the structure of the term is linear and, even better, the cost of maintaining this information, as the term is manipulated, is a constant overhead.

4 Bouncing

Given the previous discussion, equality for indefinite terms is understood and so we can focus our attention on definite terms. The main difficulty of the decision procedure concerns equality in the homset $\hom(X_0 \times X_1, A_0 + A_1)$. However, the proof of Theorem 3 has revealed an important fact: *if two terms in this homset have a definite denotation, then any path in the diagram of cardinals that witnesses the equality between them cannot cross a corner of the diagram*; that is, such a path must *bounce* backward and forward on one side:

$$\hom(X_0 \times X_1, A_j) \xleftarrow{\pi_i} \hom(X_i, A_j) \xrightarrow{\sigma_j} \hom(X_i, A_0 + A_1)\,. \tag{4}$$

In other words, in order to understand definite maps we need to study the pushouts of the above spans. Notice that the proof of Theorem 3 also reveals that some simple paths in the diagram of cardinals have bounded length. However, that proof does not provide a bound for the length of paths that bounce on one side. It is the purpose of this section to argue that such a bound does indeed exist and to explore the algorithmic consequences.

We start our analysis by considering a general span $B \xleftarrow{f} A \xrightarrow{g} C$ of sets and by recalling the construction of its colimit, the pushout $B +_A C$. This can be constructed by subdividing B and C into the image of A and the complement of that image. Thus, if $B = \mathsf{Im}(f) + B'$ and $C = \mathsf{Im}(g) + C'$ then $B +_A C = A' + \mathsf{Im}(\rho) + B'$ where $\rho : A \to B +_A C$. The image $\mathsf{Im}(\rho)$ is the quotient of A with respect to the equivalence relation witnessed by "bouncing data"; *bouncing data* is a sequence of elements of A, $(a_0, a_1, ..., a_n)$, with, for each $0 \leq i < n$ either $f(a_i) = f(a_{i+1})$ or $g(a_i) = g(a_{i+1})$. Bouncing data, $(a_0, a_1, ..., a_n)$, is said to be *irredundant* if adjacent pairs in the sequence are identified for different reasons. Thus, in irredundant bouncing data if $f(a_i) = f(a_{i+1})$ then $f(a_{i+1}) \neq f(a_{i+2})$ and similarly for g. Redundant bouncing data can always be improved to be irredundant by simply eliding intermediate redundant steps.

For bouncing data of length 2, (a_0, a_1, a_2), we shall write $a_1 : a_0 \rightsquigarrow a_2$ to indicate $f(a_0) = f(a_1)$ and $g(a_1) = g(a_2)$, and we shall call a_1 a *bouncer* from a_0 to a_2. The following is a general observation concerning pushouts of sets:

Proposition 5. *For any pushout of $B \xleftarrow{f} A \xrightarrow{g} C$ in sets the following are equivalent:*

1. *If a_0, a_n are related by some bouncing data, then they are related by bouncing data of length at most 2.*
2. *The equivalence relations generated by f and g commute.*
3. *The pushout diagram is a weak pullback, i.e. the comparison map to the pullback is surjective.*

Moreover, when one of these equivalent conditions holds, the pushout is a pullback iff for every a_0 and a_2 related by bouncing data there is a unique *element a_1 such that $a_1 : a_0 \rightsquigarrow a_2$.*

Surprisingly, this altogether special situation holds in $\Sigma\Pi(\mathbb{A})$. More precisely, we say that the homset $\hom(X_i, A_j)$ *bounces* if, for each pair of objects X_{1-i}, A_{1-j}, the span (4) has a pushout which makes the homset $\hom(X_i, A_j)$ the pullback. Intuitively, $\hom(X_i, A_j)$ bounces if,

$$\begin{tikzcd}
 & X_i \arrow[r, "f"] \arrow[ddr, dotted, "\exists!\, h"] & A_j \arrow[dr, "\sigma_j"] & \\
X_0 \times X_1 \arrow[ur, "\pi_i"] \arrow[dr, "\pi_i"'] & & & A_0 + A_1 \\
 & X_i \arrow[r, "g"'] & A_j \arrow[ur, "\sigma_j"'] &
\end{tikzcd}$$

whenever the upper and lower legs of the diagram on the right are equal (and definite), this is because of a unique bouncer $h : f \rightsquigarrow g$, where h is shown dotted and the fact that it is a bouncer means that the two smaller rectangles commute. Thus we have:

Theorem 6. *In $\Sigma\Pi(\mathbb{A})$ all homsets bounce.*

The Theorem implies that if f and g are related by a bouncing path in the diagram of cardinals, then there exists a path of length at most 2 relating them.

The proof of the Theorem 6 relies on a tricky structural induction on the pairs (X_i, A_j). Rather than presenting it here, we shall illustrate the proof for the special case of $\Sigma\Pi(\emptyset)$, the initial $\Sigma\Pi$-category. Here the situation is much simpler, as noted above, since each object is either pointed or copointed, but not both. We observe first that when there is a map from X_i to A_j, if X_i is pointed then A_j must be pointed as well and, dually, when A_j is copointed X_i must be copointed. As X_i and A_j must be either pointed or copointed it follows that X_i is pointed (respectively copointed) if and only if A_j is. However, if A_j is pointed then σ_j is monic so the bouncer h is forced to be f. Otherwise, if A_j is not pointed, then A_j is copointed and X_i as well; then π_i is epic and the bouncer h is forced to be g.

When $h \in \{ f, g \}$, say that the bouncers $h : f \rightsquigarrow g$ is trivial. While $\Sigma\Pi(\emptyset)$ has only trivial bouncers, the next example shows that this not in general the case. Let $k : x \to a$ be an arbitrary map of $\mathbb{A}$, let $X_0 = (0 \times 0) + \eta(x)$ and $A_0 = (1 + 1) \times \eta(a)$, let $z : 0 \times 0 \to 1 \times 1$ be the unique disconnect. Recalling that an arrow from a coproduct to a product might be represented as a matrix, define

$$f = \begin{pmatrix} z & \pi_0(\{\}) \\ \sigma_0(\langle\rangle) & \eta(k) \end{pmatrix} \qquad h = \begin{pmatrix} z & \pi_1(\{\}) \\ \sigma_0(\langle\rangle) & \eta(k) \end{pmatrix} \qquad g = \begin{pmatrix} z & \pi_1(\{\}) \\ \sigma_1(\langle\rangle) & \eta(k) \end{pmatrix}$$

as arrows of the homset $\hom(X_0, A_0)$. Then $h : f \rightsquigarrow g$ is an example of a non-trivial bouncer whenever X_1 is copointed and A_1 is pointed, since then f and h are coequalized by σ_0 and h and g are equalized by π_0. Notice, however, that this example relies crucially on having atomic objects. Also, this is a sort of minimal example of a non trivial bouncer; it suggested to us that the equivalence relations generated by σ_0 and π_0 might commute, see Proposition 5.

We conclude this Section by sketching an algorithm — named `equivalent`, which we present on the right for $\Sigma\Pi(\emptyset)$ – that computes whether a term f of the homset $\hom(X_0 \times X_1, A_j)$ is equivalent to a term g of the homset $\hom(X_i, A_0 + A_1)$ within the pushout of the span (4). The algorithm tries to lift f and g to f', g' in the homset $\hom(X_i, A_j)$ and, if successful, it tests for the existence of a bouncer $h : f' \rightsquigarrow g'$. Notice that the algorithm is defined by mutual recursion on the general decision procedure `equal`.

```
let equivalent f g =
  let f',g' be such that
    f ≡ σ_j(f') and g ≡ π_i(g')
  in if f',g' do not exist then
         false
     elseif X_{1-i} is copointed then
         equal σ_j(f') σ_j(g')
     else (*A_{1-j} is pointed*)
         equal π_i(f') π_i(g')
```

5 The decision procedure

We present in Figure 3 the decision procedure for $\Sigma\Pi(\emptyset)$. The general decision procedure for $\Sigma\Pi(\mathbb{A})$ – which depends on having a decision procedure for $\mathbb{A}$ – is considerably complicated by having to construct non-trivial bouncers; we describe it in the full paper.

The procedure. The procedure starts with two parallel terms in $\Sigma\Pi(\emptyset)$, $f, g : X \to A$. If X is initial or A is final then we are done – there are of course no maps if X is final and A is initial. If either X is a coproduct or A is a product we can decompose the maps and recursively check the equality of the components. Thus, if $X = X_1 + X_2$ then $f = \{\sigma_0; f, \sigma_1; f\}$ and $g = \{\sigma_0; g, \sigma_1; g\}$, and then $f = g$ if and only if $\sigma_i; f = \sigma_i; g$ for $i = 0, 1$. This requires that one cut-eliminates the compositions with σ_i – which can be performed in time linear in the size of the term.

This reduces the problem to the situation in which the domain of the maps is a product and the codomain is a coproduct. Here we have to consider two cases:

Indefinite maps. In section 3 we mentioned that in time linear on the size of the maps (which is in turn bounded by the product of the types) one can determine whether the map is pointed (and produce a point) or copointed (and produce a copoint). If both terms are pointed and copointed then they are the unique disconnect and we are done. If one term is just pointed the other must be just pointed and the points must agree (and dually for being just copointed).

Definite maps. When the maps are definite then a first goal is to determine whether the term f factors through a projection or a coprojection or, indeed, both (i.e. $f = \sigma_i(f')$ or $f = \pi_j(f')$). These factoring properties, as was discussed above, can be determined in linear time. Using these properties – remembering that a path in the diagram of cardinals that relates two definite terms can only move along a side – there are two cases, either they bounces or they do not. It they bounce we can reduce the problem to the case when one term factors

```
let equal f g = match (dom f,cod g) with
    (0,_) | (_,1) -> true
  | (1,A₀+A₁) ->
      let i,f',j,g'b be such that
        f ≡ σ_i(f') and g ≡ σ_j(g')
      in if i=j then equal f' g' else false
  | (Y₀×Y₁,0) -> ... dual
  | (_,A₀×A₁) ->
      let f₀,f₁,g₀,g₁ be such that
        f ≡ ⟨f₀,f₁⟩ and g ≡ ⟨g₀,g₁⟩
      in (equal f₀ g₀) && (equal f₁ g₁)
  | (Y₀+Y₁,_) -> ... dual
  | (X₀×X₁,A₀+A₁) ->
      if definite f g then
            match (f,g) with
                (π_i(f'),σ_j(g')) | (σ_j(g'),π_i(f')) -> equivalent f' g'
              | (π_i(f'),π_i(g')) ->
                  let i,g̃ be such that
                    π_i(g') ≡ σ_j(g̃)
                  in
                    if such i,g̃ do not exist then equal f' g'
                    else equivalent f' g̃
              | (σ_i(f'),σ_i(g')) -> ... dual
              |  _   -> false
      else equal_indefinite f g
```

Fig. 3. The decision procedure for $\Sigma\Pi(\emptyset)$.

through a projection and the other through a coprojection (using `equivalent`). If the terms do not bounce then they both must factor *syntactically* in the same manner so that f is $\sigma_0(f')$ and g is $\sigma_0(g')$, then f' must equal g'.

Complexity. To obtain the complexity of this algorithm we shall use an important observation: *in $\Sigma\Pi(\emptyset)$ the size of any cut-eliminated term representing an arrow $t : X \to A$ is bounded by the product of the sizes of the types and its height is bounded by the sum of the heights of the types.* This is proven by a simple structural induction.

The decision procedure now uses one preprocessing sweep to annotate the terms (and the types) with information concerning what is pointed and copointed. Then the main equality algorithm is applied which employs two sorts of algorithm (on subterms), which manipulate the terms and require linear time on the maximal size of the input and output terms.

The first of these algorithm simply forms a tuple when the codomain is a product and a cotuple when the domain is a sum. The second algorithm determines whether a term can be factored via a projection or coprojection and returns a factored version. Getting this to run in linear time does require that the pointed and copointed information can be retrieved in constant time (which is managed by preprocessing the terms).

The other major step in the algorithm, which we have not discussed for the general case, involves finding a bouncer. In the $\Sigma\Pi(\emptyset)$ case this involves determining which of the projection or coprojection is respectively epic or monic. This, in turn, is determined by the pointedness or copointedness of the components of the type which can usefully be calculated in the preprocessing stage – and so is constant time.

Essentially this means that the algorithm at each node of the term requires processing time bounded by a time proportional to the (maximal) size of the subterm. Such a pattern of processing is bounded by time proportional to the height of the term times the size. We therefore have:

Proposition 7. *To decide the equality of two parallel terms $t_1, t_2 : A \to B$ in $\Sigma\Pi(\emptyset)$ has complexity in $\mathcal{O}((\mathsf{hgt}(A) + \mathsf{hgt}(B)) \cdot \mathsf{size}(A) \cdot \mathsf{size}(B))$.*

The analysis of the algorithm for $\Sigma\Pi(\mathbb{A})$ is more complex and is left to the fuller exposition.

References

1. Lambek, J.: Deductive systems and categories. I. Syntactic calculus and residuated categories. Math. Systems Theory **2** (1968) 287–318
2. Došen, K.: Cut elimination in categories. Volume 6 of Trends in Logic—Studia Logica Library. Kluwer Academic Publishers, Dordrecht (1999)
3. Cockett, J.R.B., Seely, R.A.G.: Finite sum-product logic. Theory Appl. Categ. **8** (2001) 63–99 (electronic)
4. Hu, H., Joyal, A.: Coherence completions of categories. Theoret. Comput. Sci. **227**(1-2) (1999) 153–184 Linear logic, I (Tokyo, 1996).
5. Hughes, D.J.D., van Glabbeek, R.J.: Proof nets for unit-free multiplicative-additive linear logic (extended abstract). In: LICS, IEEE Computer Society (2003) 1–10
6. Došen, K., Petrić, Z.: Bicartesian coherence revisited. In Ognjanovic, Z., ed.: Logic in Computer Science, Zbornik Radova. Volume 12. (2009) arXiv:0711.4961.
7. Blass, A.: A game semantics for linear logic. Ann. Pure Appl. Logic **56**(1-3) (1992) 183–220
8. Joyal, A.: Free lattices, communication and money games. In: Logic and scientific methods (Florence, 1995). Volume 259 of Synthese Lib. Kluwer Acad. Publ., Dordrecht (1997) 29–68
9. Santocanale, L.: Free μ-lattices. J. Pure Appl. Algebra **168**(2-3) (2002) 227–264 Category theory 1999 (Coimbra).
10. Cockett, J.R.B., Pastro, C.A.: A language for multiplicative-additive linear logic. Electr. Notes Theor. Comput. Sci. **122** (2005) 23–65
11. Cockett, J.R.B., Pastro, C.A.: The logic of message passing. CoRR **abs/-math/0703713** (2007)
12. Mac Lane, S.: Categories for the working mathematician. Second edn. Volume 5 of Graduate Texts in Mathematics. Springer-Verlag, New York (1998)
13. Joyal, A.: Free bicomplete categories. C. R. Math. Rep. Acad. Sci. Canada **17**(5) (1995) 219–224
14. Joyal, A.: Free bicompletion of enriched categories. C. R. Math. Rep. Acad. Sci. Canada **17**(5) (1995) 213–218